ILMATIETEEN LAITOS
METEOROLOGISKA INSTITUTET
FINNISH METEOROLOGICAL INSTITUTE

Automatic Image Analysis – Day 1

Mikko Syrjäsuo

Earth Observation

Finnish Meteorological Institute

Introduction to this lecture series

- A selection of topics is discussed
  - **terminology**
  - file formats and other practical aspects
  - the exercises are examples of real life problems
- Notable omissions:
  - transforms such as 2D FFT or DWT
  - image reconstruction
  - time-series of images (i.e. motion analysis)
  - noise reduction

Introduction

“A picture is worth a thousand words”

We use vision to process approximately 75% of the incoming information

⇒

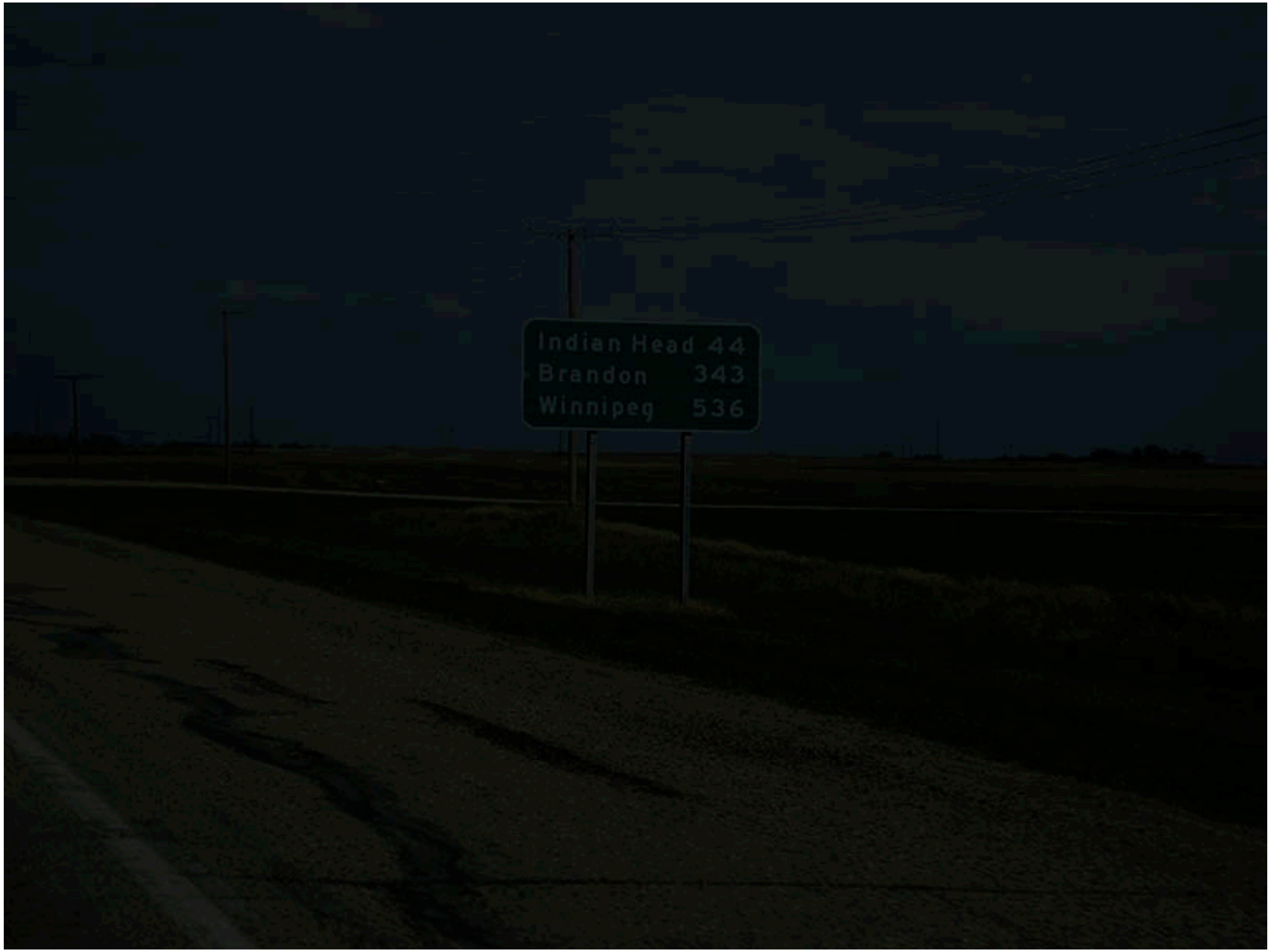
Automated image processing is invaluable when processing large amounts of image data

Objectives – image enhancement

(Automatic) image processing to improve the image to aid the (human) analysis system in interpretation

Input image ⇒ output image

Terminology: point processing, spatial processing, image enhancement, geometric corrections, noise reduction


Indian Head 44
Brandon 343
Winnipeg 536

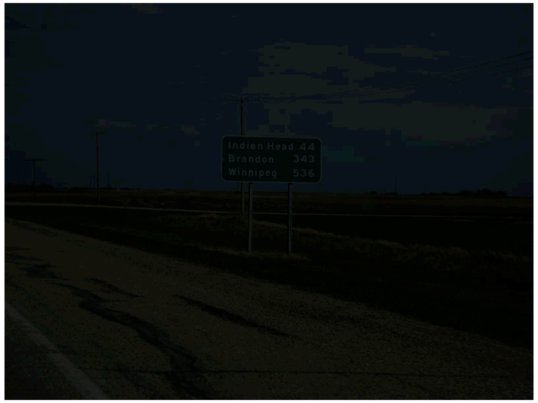

Indian Head 44
Brandon 343
Winnipeg 536

Objectives – image analysis

(Automatic) image processing for analysing the image contents (automatically)

Input image ⇒ something else

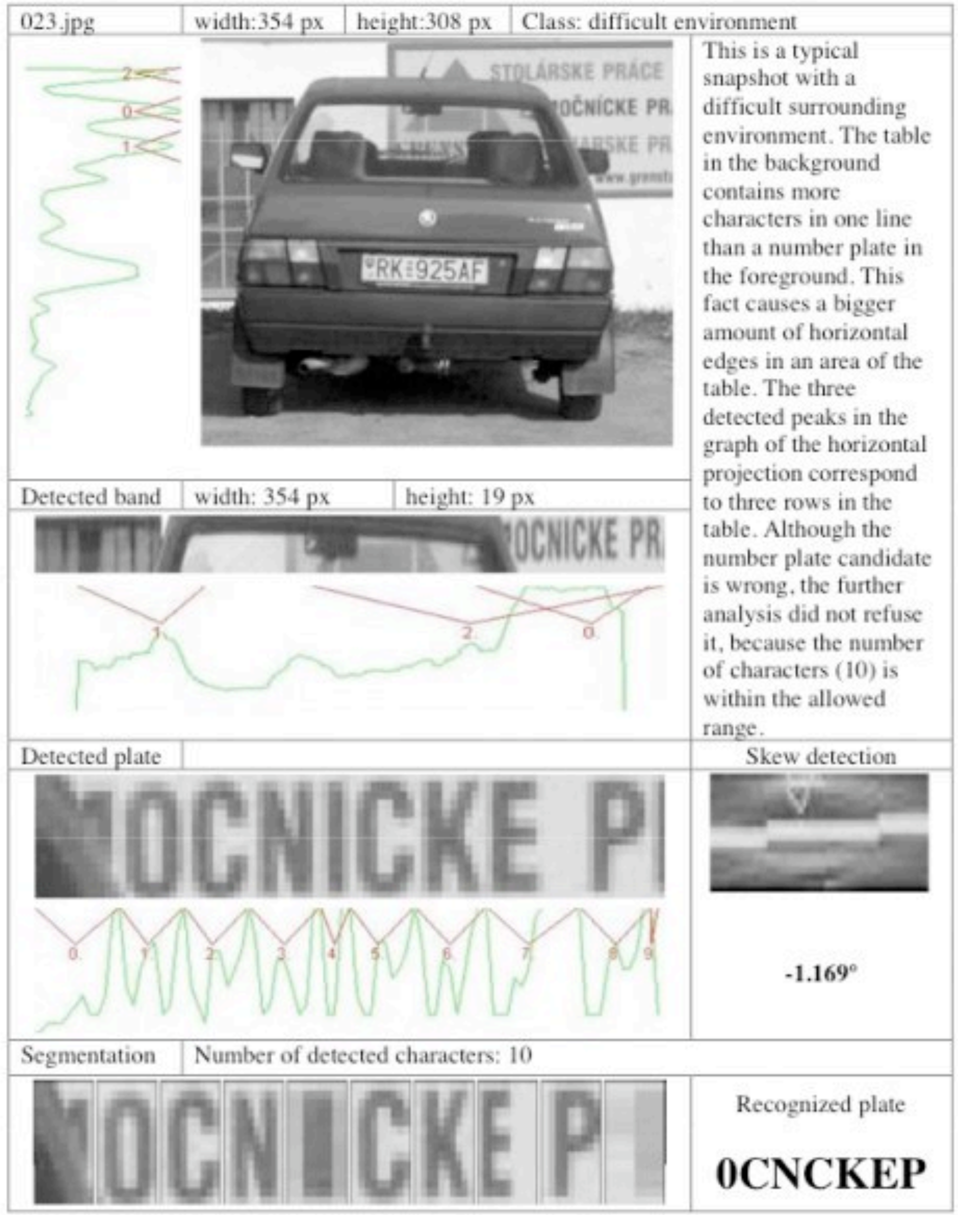
Terminology: pattern recognition, computer vision, image understanding, statistical/machine learning



Pre-processing is typically required before successful analysis is possible!

ILMATIETEEN LAITOS
METEOROLOGISKA INSTITUTET
FINNISH METEOROLOGICAL INSTITUTE

Ondrej Martinsky, "*Algorithmic and mathematical principles of automatic number plate recognition systems*", B.Sc. thesis, Brno University of Technology, 2007

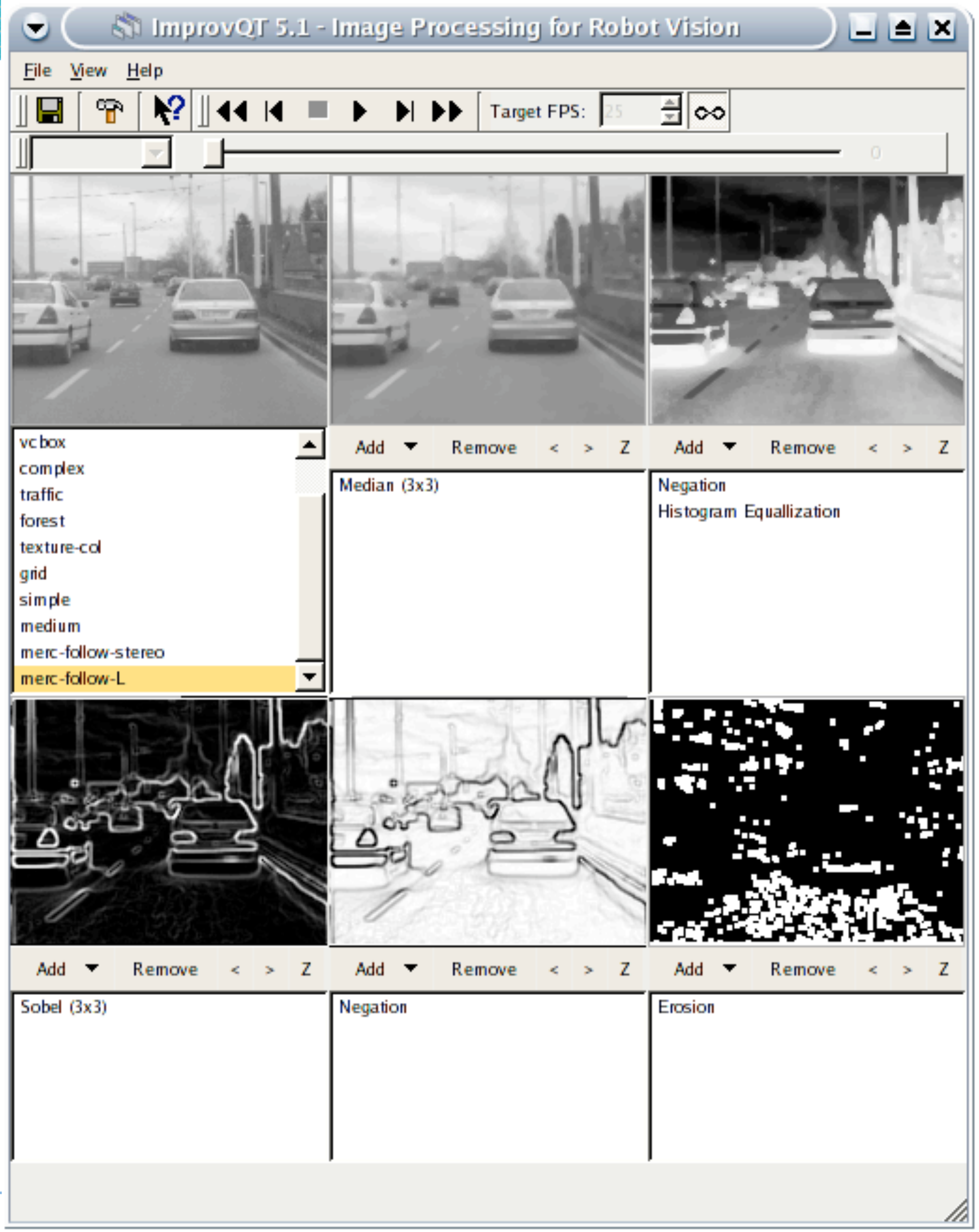


ILMATIETEEN LAITOS
METEOROLOGISKA INSTITUTET
FINNISH METEOROLOGICAL INSTITUTE

Image processing for robot vision

Improv is a tool for basic real time image processing at low resolution, e.g. suitable for mobile robots.It has been developed for PCs with the *Linux* operating system and runs under X Windows. *Improv* works with a number of inexpensive low-resolution digital cameras (no framegrabber required).

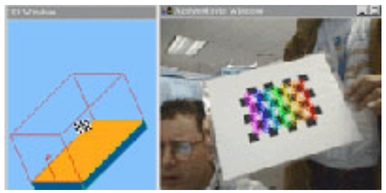
http://robotics.ee.uwa.edu.au/improv/



Open Source Computer Vision Library

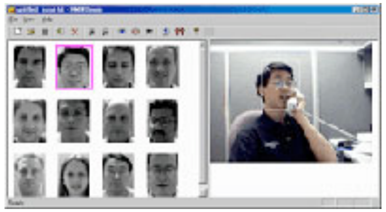
- http://www.intel.com/technology/computing/opencv/
- an open source library optimised for Intel processors

Camera calibration



Tracking

Recognition



Objectives – image compression

Input image ⇔ compressed image

Transmission and storing images

Terminology: information preserving i.e. lossless compression vs. lossy compression, entropy, compression ratio

Objectives – automatic image analysis

Computer vision

- qualitative/quantitative explanation of images
- structural/statistical recognition of objects

1. image preprocessing
2. feature extraction
   - **features are normally multidimensional!**
3. object recognition
   - **use clustering, learning, adaptive models, etc.**
4. image/scene understanding

Overlap with other fields

- Pattern recognition
  - **multidimensional data**
- Artificial intelligence
- Observational psychology
  - human information processing system
- Computer graphics
- Optics
- Signal processing

Applications

- Military
- Computer industry (games, software)
- Remote sensing
- Medical imaging
- Industrial applications (machine vision)
- Image transmission and archives
- Consumer electronics (digital cameras, mobile phones)
- Science applications (e.g. astronomy, space physics)

Image sources – different wavelengths

- Gamma-ray (medicine, astronomy)
- X-ray (medicine, planetology)
- UV (microscopy, astronomy)
- Visible light (family portraits, security applications)
- Infrared (remote sensing)
- Microwave and other radio wavelengths (radars, medicine)
- Seismology (100Hz)
- Ultrasound (medicine)
- Simulation outputs and other **computationally generated** images

Day One – doing it manually

- Introduction
- Overview of common image formats and tools for interactive image processing
  - **Exercise #1 (image enhancement)**
- Human vision peculiarities
- Practical issues
  - **Exercise #2 (camera calibration)**

Day Two – letting the computer do it

- Segmentation
- Image features
- Object classification and detection
  - **Exercise #3 (automatic image analysis)**

Image properties

- a multidimensional array of picture elements or **pixels**
- channel = one 2D image slice of a multidimensional array
  - **“bits per pixel/channel”**
- meta-data (e.g. information in file header, EXIF-data)

- pixel coordinates
- neighbourhood
- histogram

Image matrix

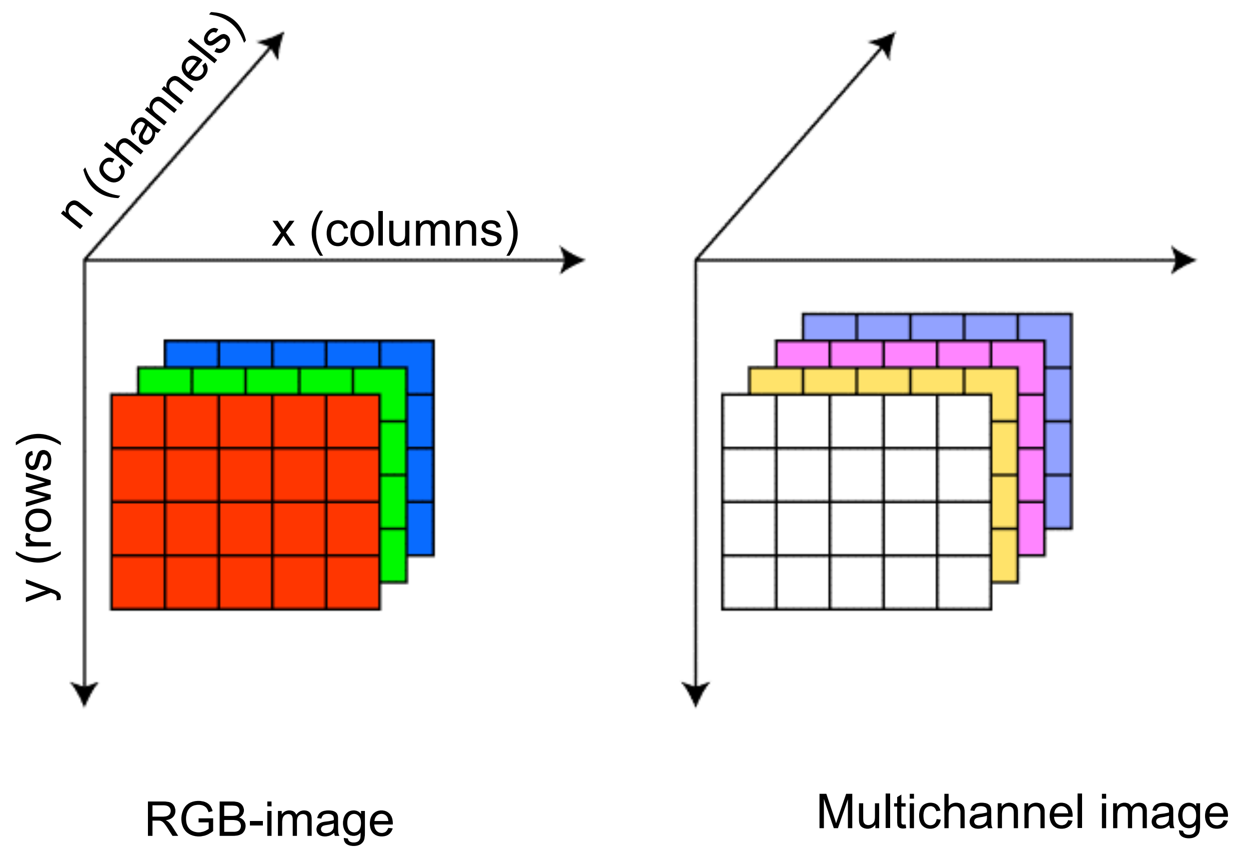
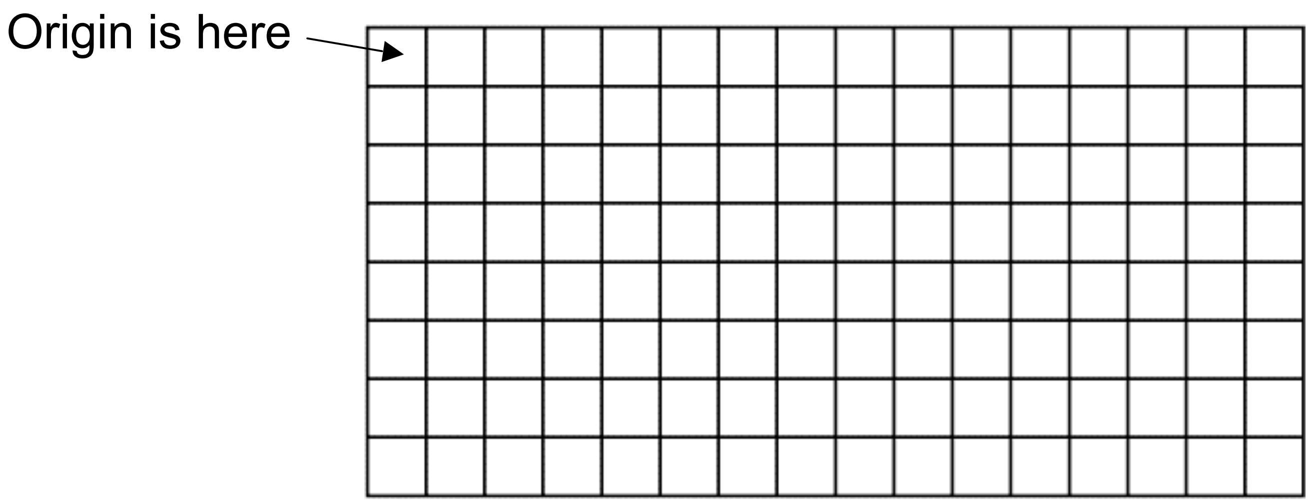


Image – pixel rows and columns



8 rows, 16 columns ⇒ 8 by 16 matrix ⇒ 16 by 8 image...

Pixel processing

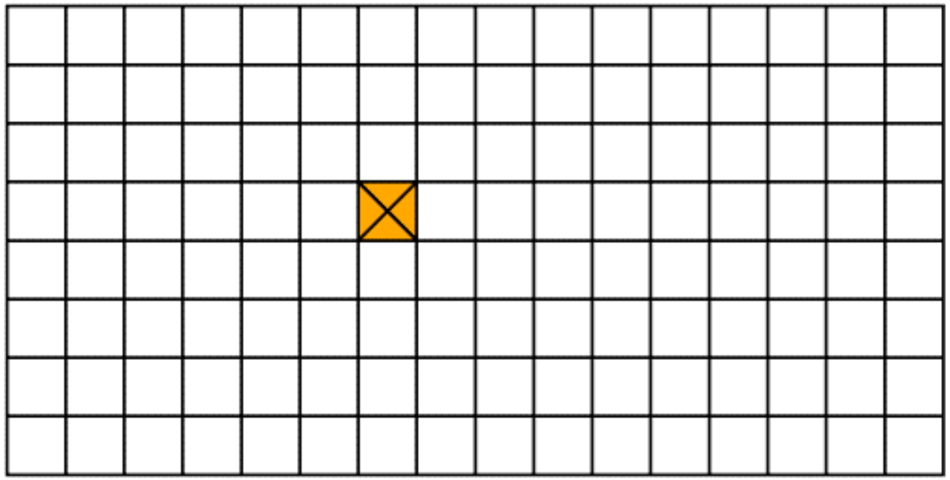


Image processing (enhancement) is normally performed by calculating a new value (to output image) from the input image pixel value ⇒ point processing

If neighbouring pixel values are also used ⇒ spatial processing

Image neighbours – 4-neighbourhood

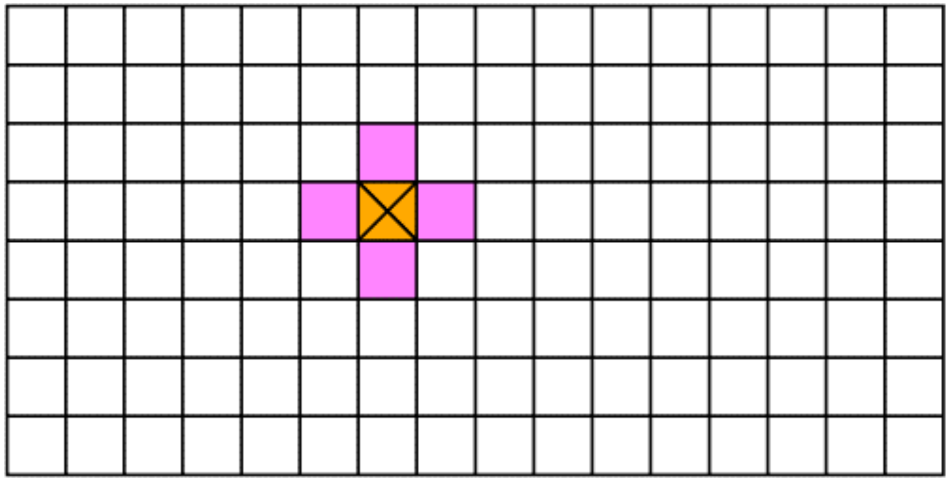


Image neighbours – 8 neighbourhood

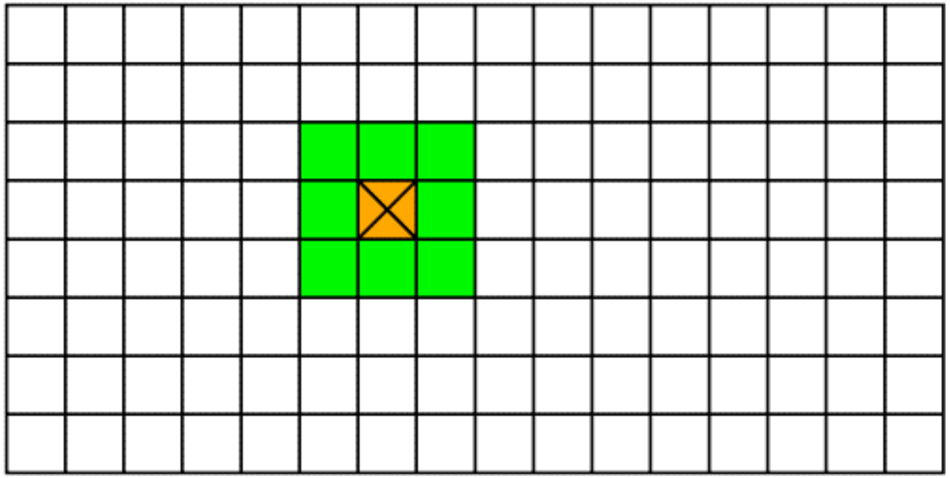
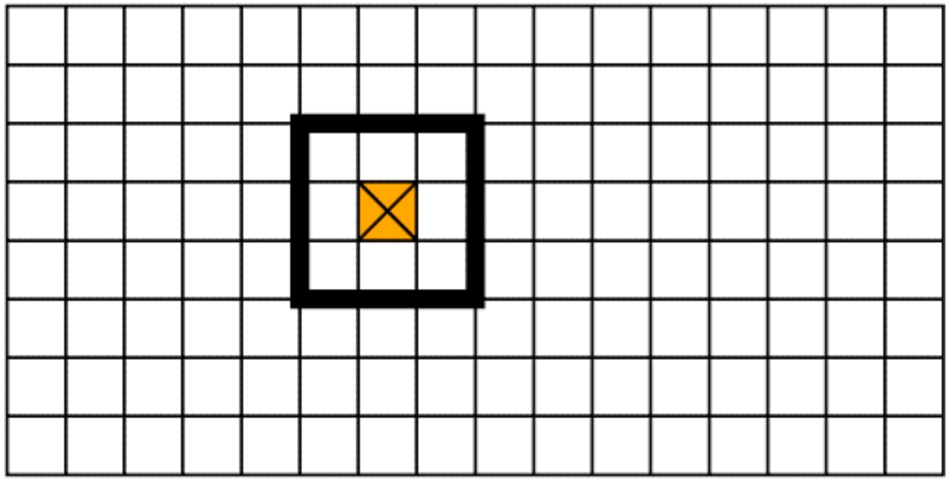


Image mask – spatial processing

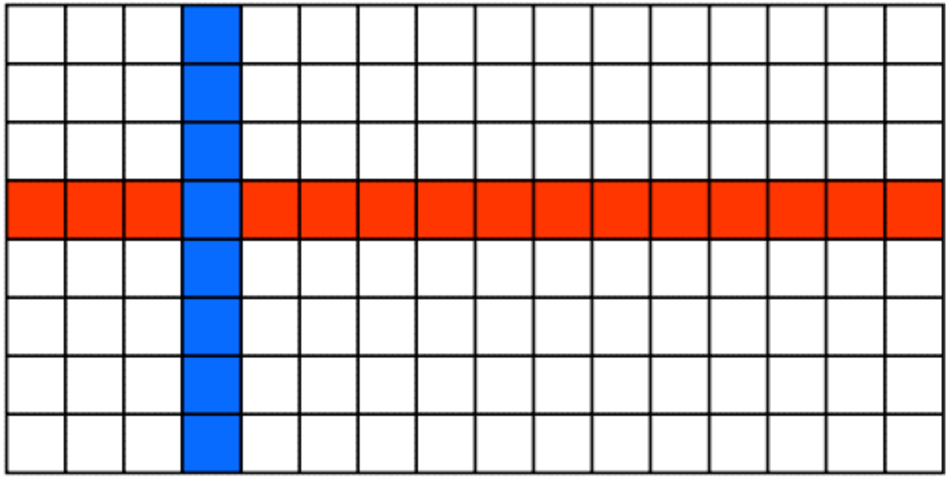


A common spatial mask uses nine pixel values to calculate the value in the output image.

What do you do if part of the mask is outside of the image?

ILMATIETEEN LAITOS
METEOROLOGISKA INSTITUTET
FINNISH METEOROLOGICAL INSTITUTE

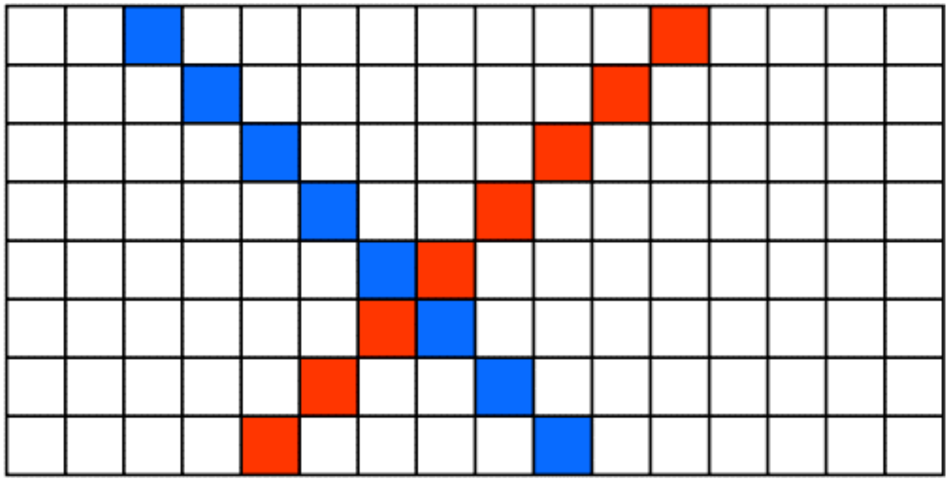
Intersecting lines



The pixel nature of images (and other sparse data) create situations where mathematical truths conflict with real life:

the lines clearly have an intersection (i.e. a common pixel)

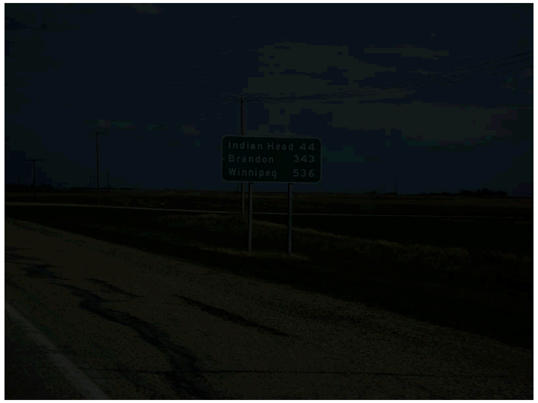
Intersecting lines?



In 2D, these two lines should have an intersection but, in this image, they do not!

Similar effects can be observed in “areas” (number of pixels inside an area) and other mathematically exact concepts. You have been warned...

Image histogram



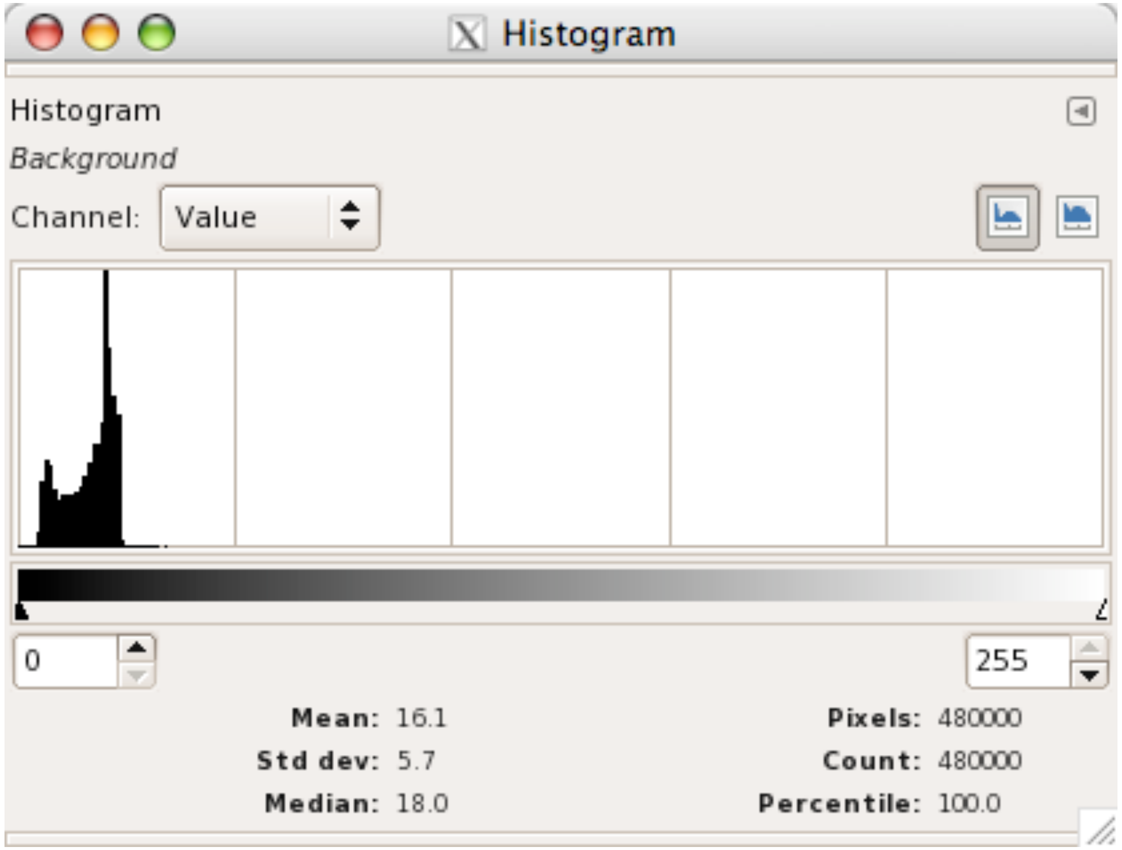


Image histogram

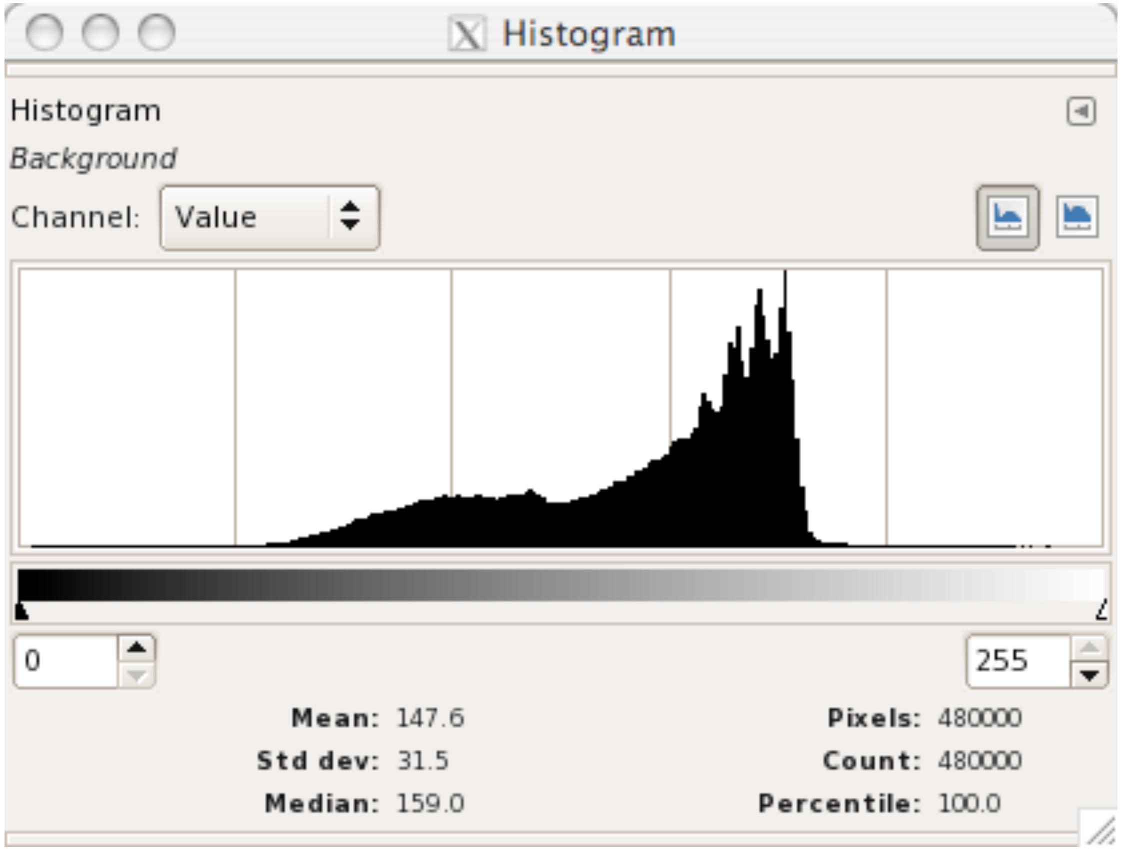


Image curves

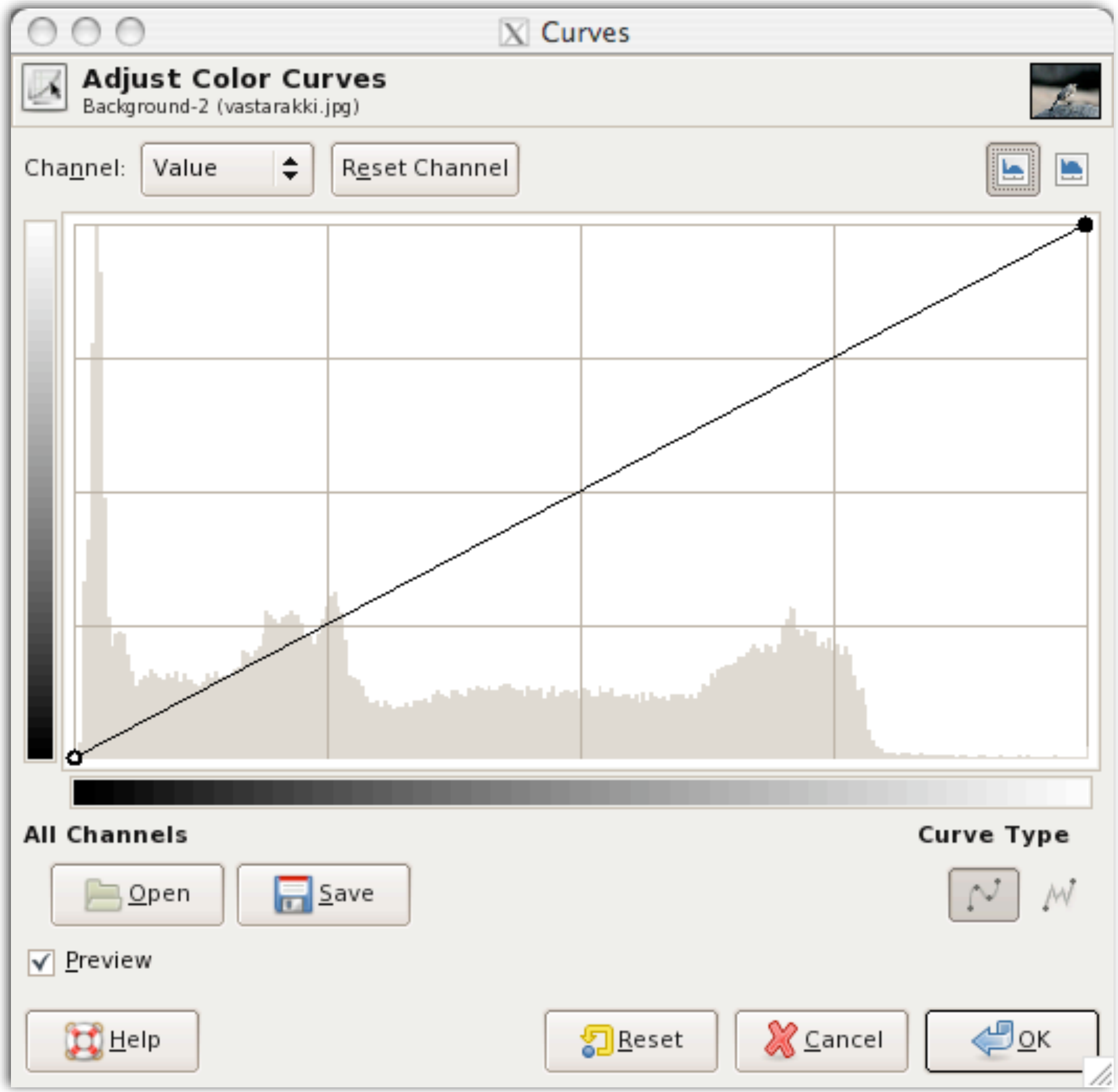


Image curves

$y=f(x)$

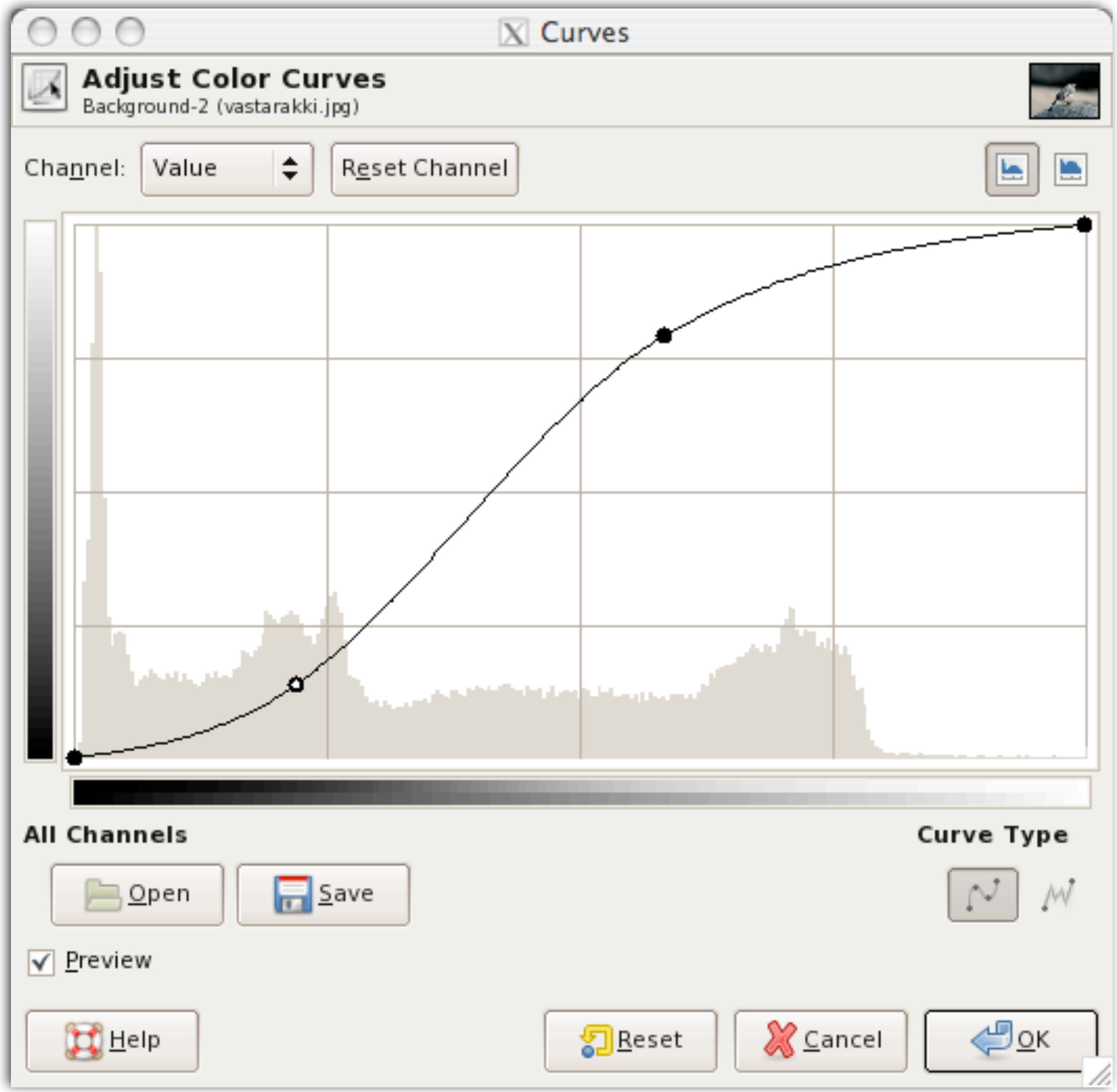


Image compression

Lossy compression (20kB = 11%)

Lossy compression (8kB = 4%)

Original image 300×200 pixels with 24-bit RGB

⇓

Uncompressed file 180kB

Image file formats (1)

- PNM (Portable AnyMap)
  - PPM (Portable PixelMap)
  - PGM (Portable GrayMap)
  - PBM (Portable BitMap)
  - PAM (Portable Arbitrary Map)
- **Simple**, no compression (use *gzip* or *bzip2*)
- Can contain human readable meta-data
- NetPBM-library + utilities ***(http://sourceforge.net/projects/netpbm)***

Excellent tools for scripts!

```
pngtopnm input.png | pnmscale 0.3 | pnmflip -tb | pnmnorm | pnmtojpeg > output.jpg
```

Image file formats (2)

- PNG (Portable Network Graphics) “ping”
  - “Replacement” of GIF-format (no patent issues!)
  - **Lossless** compression of image data
  - http://www.libpng.org/pub/png/
- TIFF (Tagged Image File Format)
  - flexible and complex
  - various compression options available (lossy/lossless)
  - under control by Adobe Systems, no major updates since 1992

Image file formats (3)

- JPEG (Joint Photographic Expert Group)
  - normally **lossy compression**
    - based on 2D Discrete Cosine Transform
    - utilises human vision by reserving more bandwidth for brightness information than colour information
  - some variation between implementations
  - additions to original file format
    - **EXIF** (digital camera meta-data)
    - **ICC** (colour profile information)
  - **8-bits per colour channel**

Image file formats (4)

- JPEG-2000
  - the successor of JPEG with many improvements (Web)
  - **lossy and lossless compression**
    - lossy compression uses Discrete Wavelet Transform with better performance than DCT
  - Region-of-Interest
    - some areas can be compressed with less loss in details
  - also 16-bits per channel can be used

Image file formats (5)

- **FITS** (Flexible Image Transport System)
  - "Astronomers' choice"
  - plain text meta-data (i.e. human readable)
  - Not all programs support reading/writing all types of FITS files
    - *http://heasarc.nasa.gov/docs/heasarc/fits.html*
  - Related science data formats
    - *Common Data Format* (CDF-file)
    - *Hierarchical Data Format* (HDF-file)
    - *Network Common Data Form* (NetCDF-file)

Image file formats (6)

- **“Binary” formats**
  - Camera raw (digital cameras)
    - often proprietary
  - Photoshop (.psd)
  - Matlab/IDL/Octave “save” files

Image import/export (1)

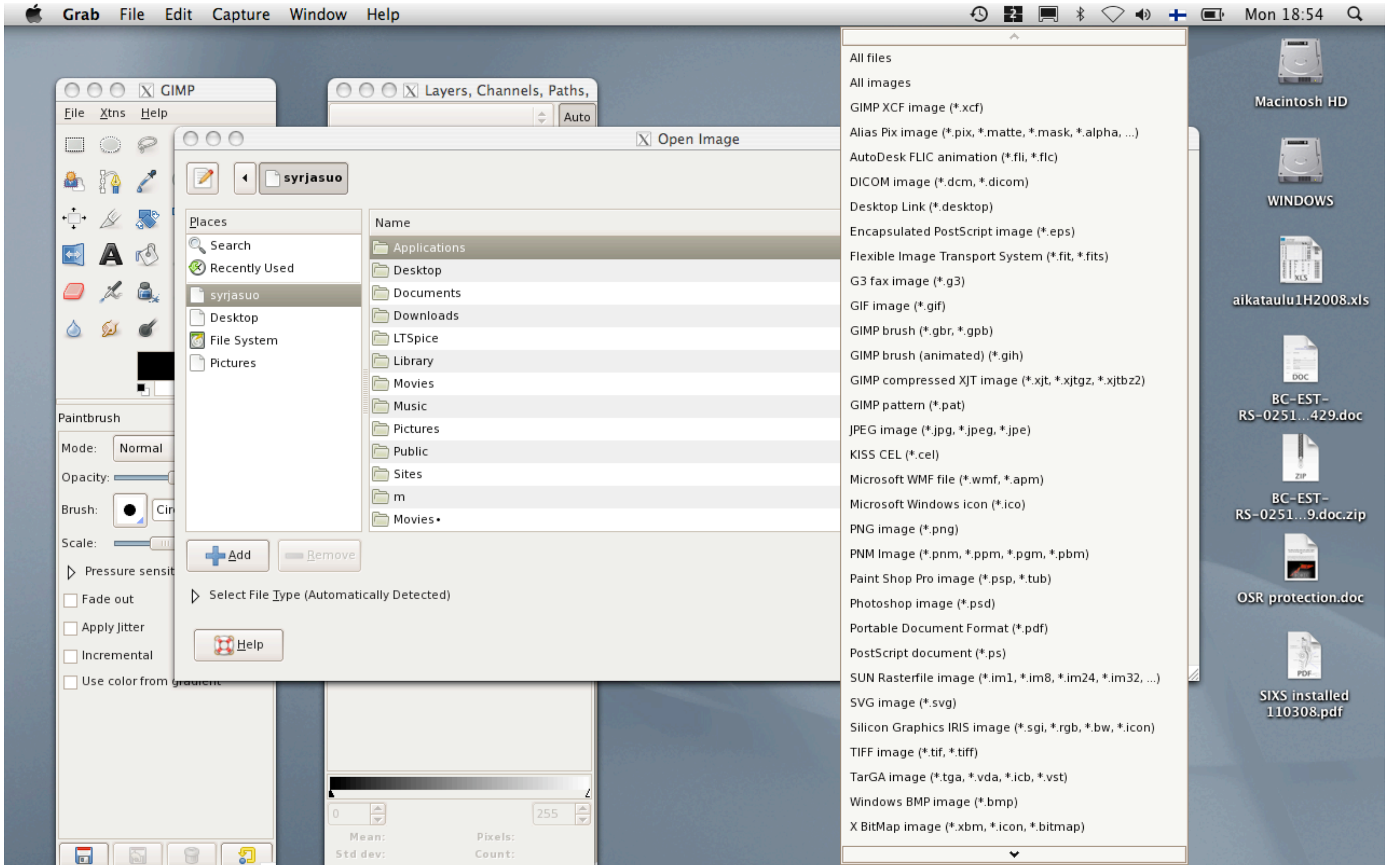
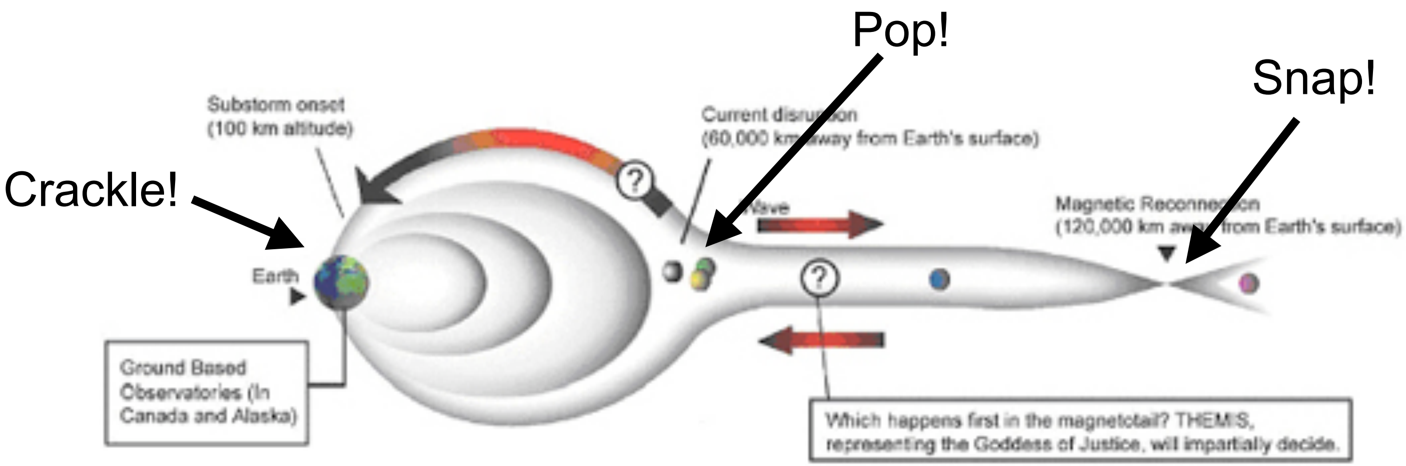


Image import/export (2)

- netpbm-library & tools
- cjpeg/djpeg/rdjpgcom/wrjpgcom
- ImageMagick
  - convert/identify
- dcraw (camera raw)
- http://freeimage.sourceforge.net
- Matlab/IDL/etc.
- Photoshop/GIMP/PaintShop Pro etc.

Exercise #1: point processing

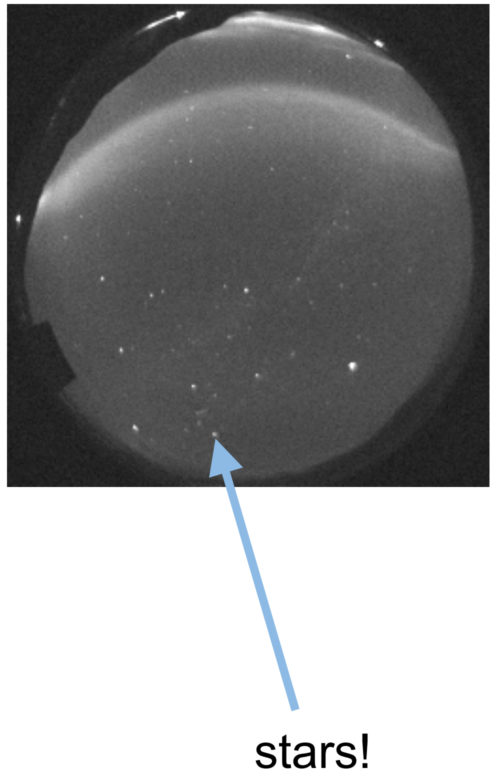
- THEMIS (Time History of Events and Macroscale Interactions during Substorms)
  - NASA mission with five satellites and 20 ground stations with magnetometers and auroral all-sky imagers
  - 110 million images per year...



Exercise #1: point processing

This is an auroral all-sky image from Whitehorse, Yukon, Canada. The fish-eye lens captures the whole night sky (clouds, stars, auroras etc.)

You will be using raw data from the THEMIS Ground-Based Observatory (GBO) in Rankin Inlet, Nunavut, Canada. Your task is to convert the input image into a JPEG-image in which stars are clearly visible.



Practical issues

- Imaging hardware
  - **mapping a 3D world into a 2D image**
  - **lens distortions**

- Human vision system
  - **colours, perception**

3D to 2D and other difficulties

- the imaging hardware maps a 3D scene into a 2D image
  - **no inverse mapping exists!!**
  - **objects occlude themselves and each other**
- complicated correspondence
  - **what does a measured intensity tell you about the scene?**
- human concepts
  - **what is "a chair"?**

Solutions (or strategies...)

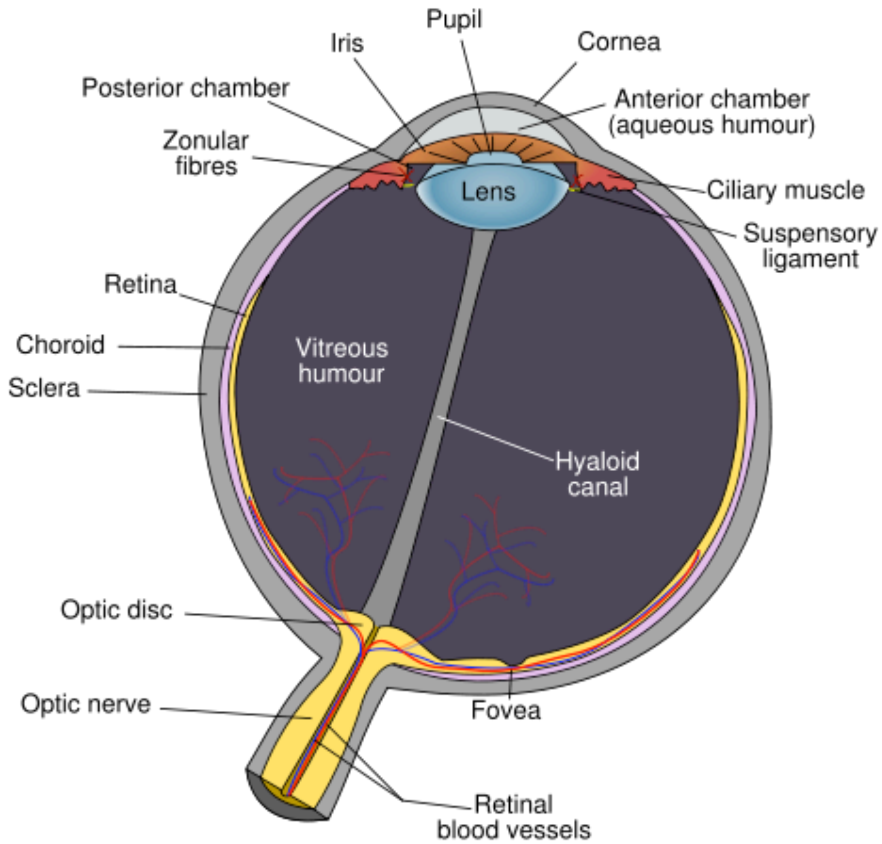
- bottom-up reconstruction
  - **the most general solution**
  - **however, may also not work at all!!**

- top-down recognition
  - **model-based**
  - **a special solution for a specific problem**

- human-in-the-loop systems

Human vision

- fovea – fine details
  - **6-7 million cones mostly in fovea**
  - **colour vision**
  - ***photopic vision***
- retina – overall picture
  - **75-150 million rods**
  - **greyscale**
  - ***scotopic vision***



Human vision – properties

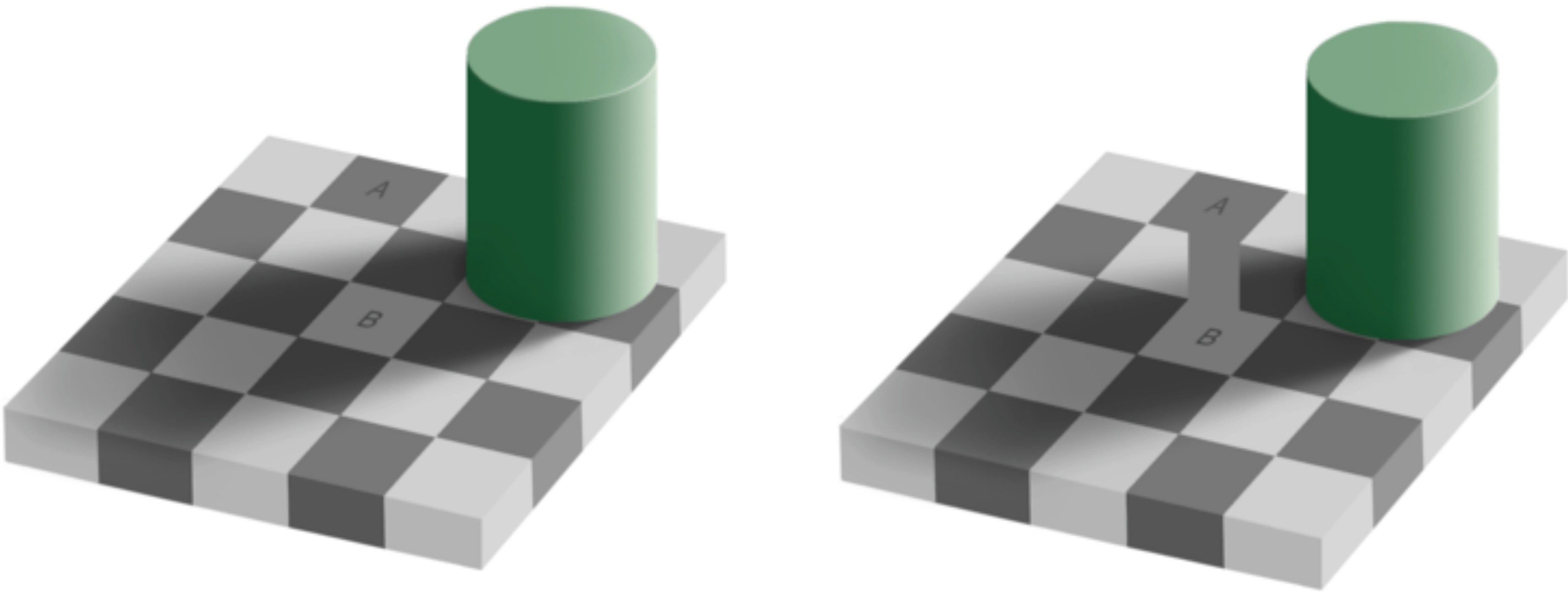
- Enormous light intensity range on the order 10^{10}
  - **brightness adaptation**
- Brightness perception is complicated
  - **“Mach bands”, “simultaneous contrast”**

Human vision – colour perception

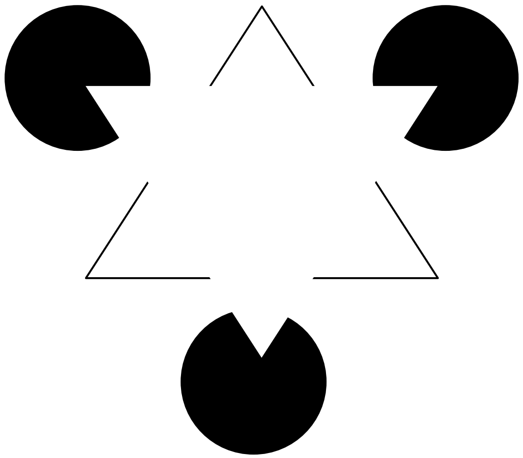
- the perception of colour depends on the surroundings
  - **“white balance” setting in digital cameras**
- individuals do not perceive colours in the same way
  - **colour blindness**

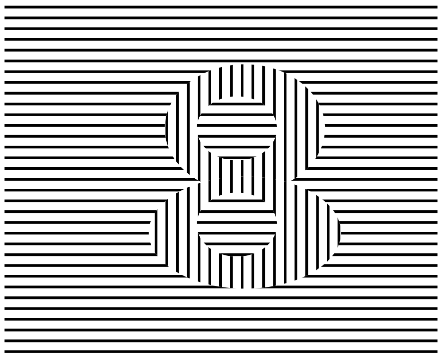
- “Wrong” colours
  - **chrominance adaptation ⇐ human vision**
  - **truly different colours (spectra) ⇐ hardware issues**

Adelson’s checker shadow illusion



Human vision – seeing boundaries





Colour recording and re-production

- Input device
  - **camera (film, electronic)**
  - **spectrometer**
- Output device
  - **monitor**
  - **printer**

Colour theory

- colour is the *perceptual* result of light incident upon the retina
- in colourimetry, the colours are specified numerically
  - **same conditions ⇒ same numbers (colour match)**
  - **stimuli that look alike ⇒ same numbers**
  - **numbers are continuous functions of physical parameters defining the spectral radiant power**
- trichromatic generalisation
  - **over a wide range of observation conditions, many colour stimuli can be matched by additive mixtures of three fixed primary stimuli**

Colour spaces – device independent

- **CIE XYZ, xyY**
  - based on colour matching functions
- **CIE Lab ($L^*a^*b^*$)**
  - more uniform colour differences
- **CIE Luv (Lu'v')**
  - more uniform brightness

There is no generally accepted "best" space.

Colour spaces – device dependent

- **ALL implementations of ALL colour recording and producing systems**
  - the same colour specification numbers may not correspond to the same input or output stimuli
- **individual spectral lines are not (exactly) re-producible**
- **the primary stimuli vary between devices**
  - colour calibration can be used to minimise perceptual errors (colour correction)
- **sRGB - standard monitor colours space**
  - nonlinear conversion from CIE XYZ

Colour horseshoe

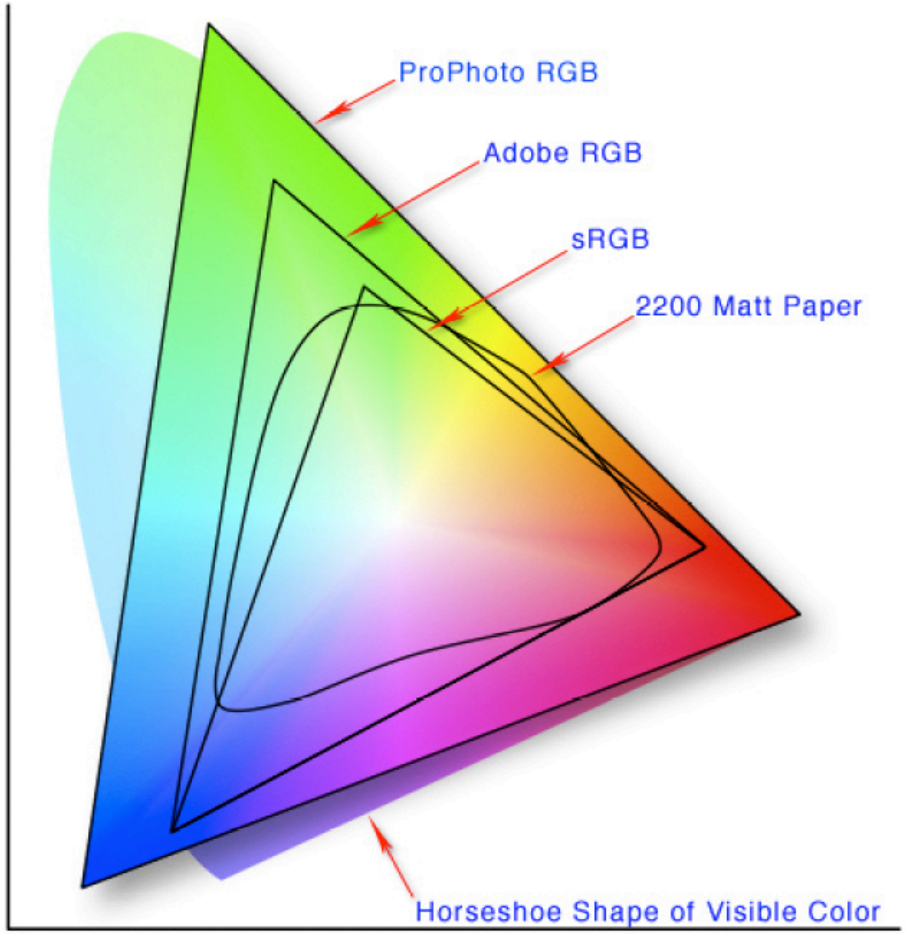


Illustration by Jeff Schewe

Colour spaces – what do you need to know?

- Colour specifications are based on experiments with a number of human observers
  - **spectra are mapped into three values that specify the perceived colour**
- Processing colour images:
  - **RGB does not match the human colour perception**
  - **compute colour differences in CIE Lab or Luv**
- The range of possible colours (gamut) depends on the selected colour space
  - **transformation between spaces may lose information**

Camera calibration

- measuring distances etc. requires knowledge about the optical system
- in calibration, the parameters of the projection from 3D to 2D are determined
  - **an incoming light ray from a direction (α,β) will illuminate pixel at (x,y)**
  - **mapping from "world coordinates" to "camera coordinates"**
- requires knowledge of the input
  - **astronomers commonly use stars as reference points**
  - **industrial settings: a target image**

Exercise #2: camera calibration

Using the star image from Rankin Inlet (exercise #1), derive the calibration parameters (k,θ,x_0,y_0) for the camera.

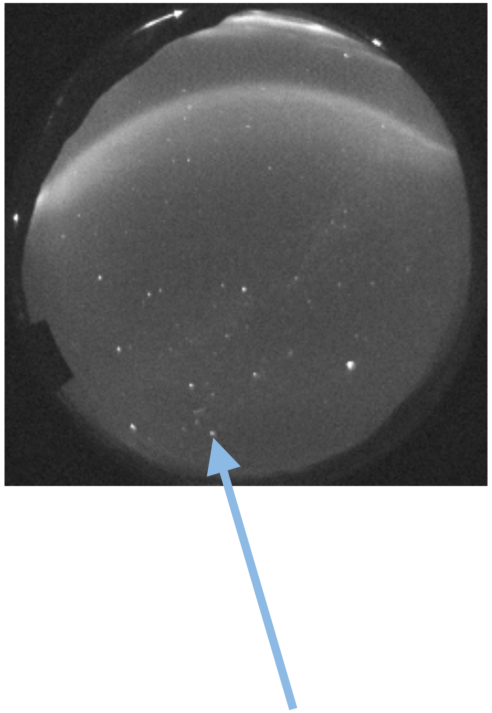
k	lens coefficient
θ	image rotation
x_0,y_0	zenith location

Assumptions:

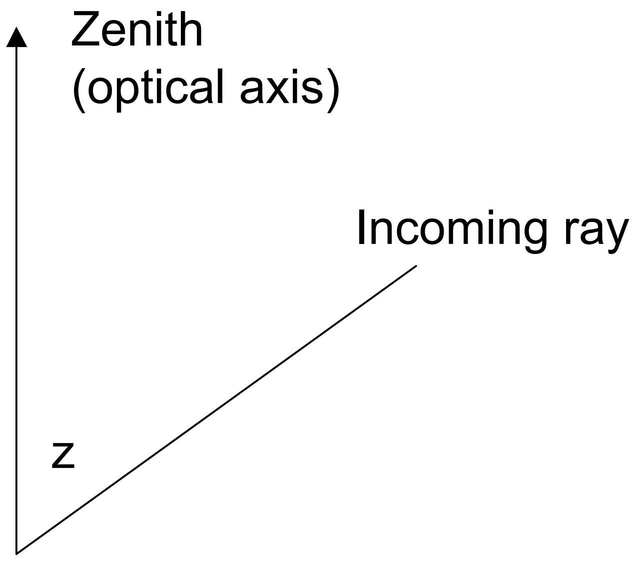
The fish eye lens projection is

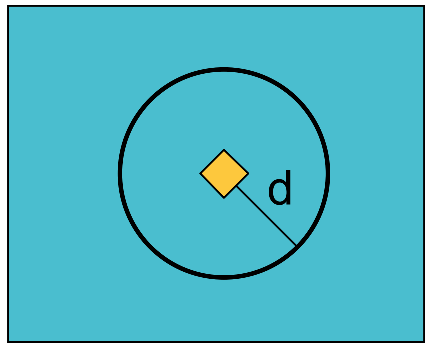
$$d=kz$$

where z is the zenith angle of the incoming ray and d is the pixel distance from the optical axis in the image



Exercise #2: fish-eye lens with equidistant projection





Constant zenith angle z = circle with a radius d

Exercise #2: camera calibration

Use stars as reference points

1. Use xephem or Stellarium etc. to find elevation (90°- zenith angle) and azimuth (rotation from north)
2. Determine the pixel coordinates of those stars
3. Use the known star locations as reference points and fit the data to the lens projection model